Distributed Representations of Words and Phrases and their Compositionality

Tomas Mikolov
Google Inc.
Mountain View
mikolov@google.com

Ilya Sutskever
Google Inc.
Mountain View
ilyasu@google.com

Kai Chen
Google Inc.
Mountain View
kai@google.com

Greg Corrado
Google Inc.
Mountain View
gcorrado@google.com

Jeffrey Dean
Google Inc.
Mountain View
jeff@google.com

张老师
2020|06|14

贪心科技 | 让每个人享受个性化教育服务

对skip-gram模型优化扩展

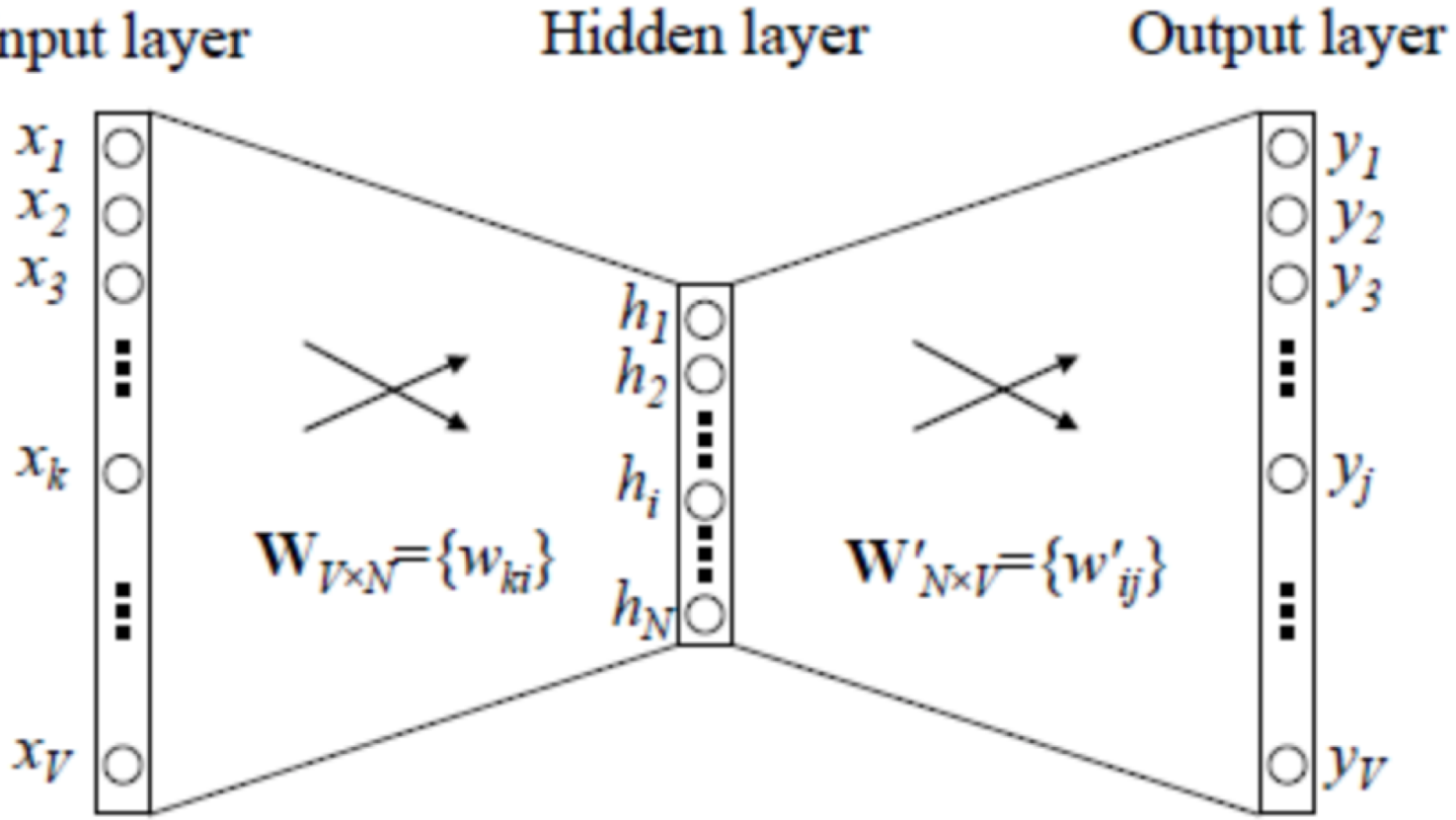
- 对高频词重采样
- Negative sampling
- 从基于单词的模型扩展到基于短语的模型

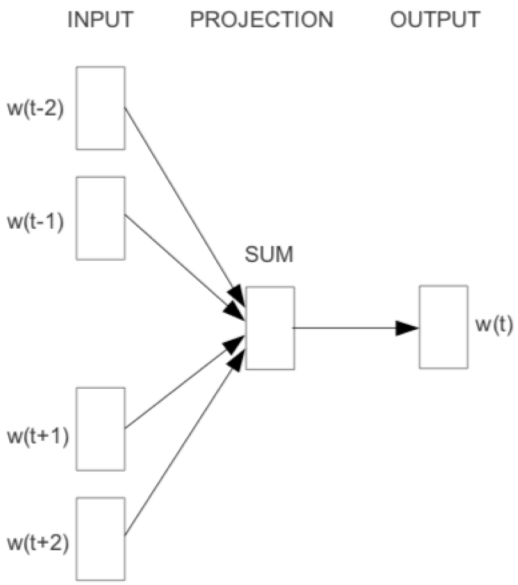
Embedding相关文章

- Word2vec：
- Efficient Estimation of Word Representations in Vector Space https://arxiv.org/pdf/1301.3781.pdf
- Distributed Representations of Words and Phrases and their Compositionality http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf
- Word2vec Parameter Learning Explained https://arxiv.org/abs/1411.2738
- https://github.com/tmikolov/word2vec

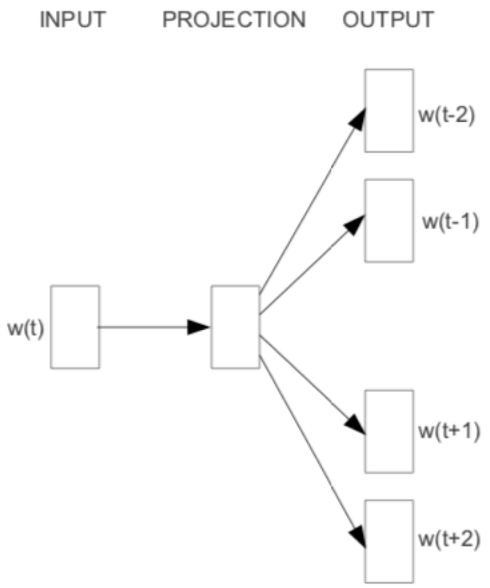
Embedding相关文章

- Word2vec的衍生及应用：
- [Item2Vec] Item2Vec-Neural Item Embedding for Collaborative Filtering (Microsoft 2016)
- [Airbnb Embedding] Real-time Personalization using Embeddings for Search Ranking at Airbnb (Airbnb 2018)
- Graph embedding：
- [DeepWalk] DeepWalk- Online Learning of Social Representations (SBU 2014)
- [LINE] LINE - Large-scale Information Network Embedding (MSRA 2015)
- [Node2vec] Node2vec - Scalable Feature Learning for Networks (Stanford 2016)


Input layer
Hidden layer
Output layer
x_1
x_2
x_3
x_k
x_V
h_1
h_2
h_i
h_N
y_1
y_2
y_3
y_j
y_V
$\mathbf{W}_{V\times N}=\{w_{ki}\}$
$\mathbf{W}'_{N\times V}=\{w'_{ij}\}$



CBOW



Skip-gram

- CBOW：continuous **bag-of-words 无序平等**
- Skip-gram

模型动机：每个词都决定了相邻的词
产生模型正样本：长度为2c+1的滑动窗口
优化目标：样本的条件概率之积最大

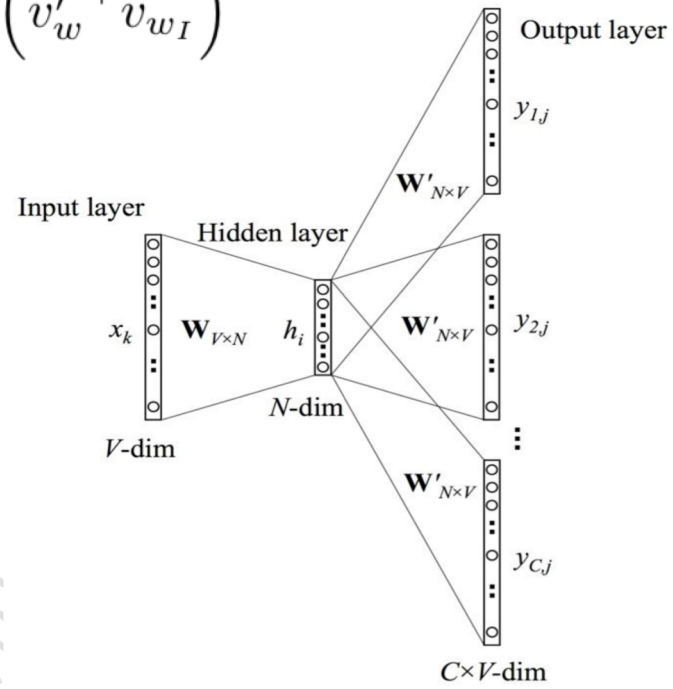
- Skip-gram训练目标：预测文本中某个词周围可能出现的词

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0} logP(w_{t+j}|w_t)$$

- C：上下文窗口大小

多分类，softmax，向量之间越相似，点乘结果越大，从而归一化后得到的概 率值也越大

$$p(w_O|w_I)=\frac{\exp\left({v'_{w_O}}^{\top} v_{w_I}\right)}{\sum_{w=1}^{W}\exp\left({v'_w}^{\top} v_{w_I}\right)}$$

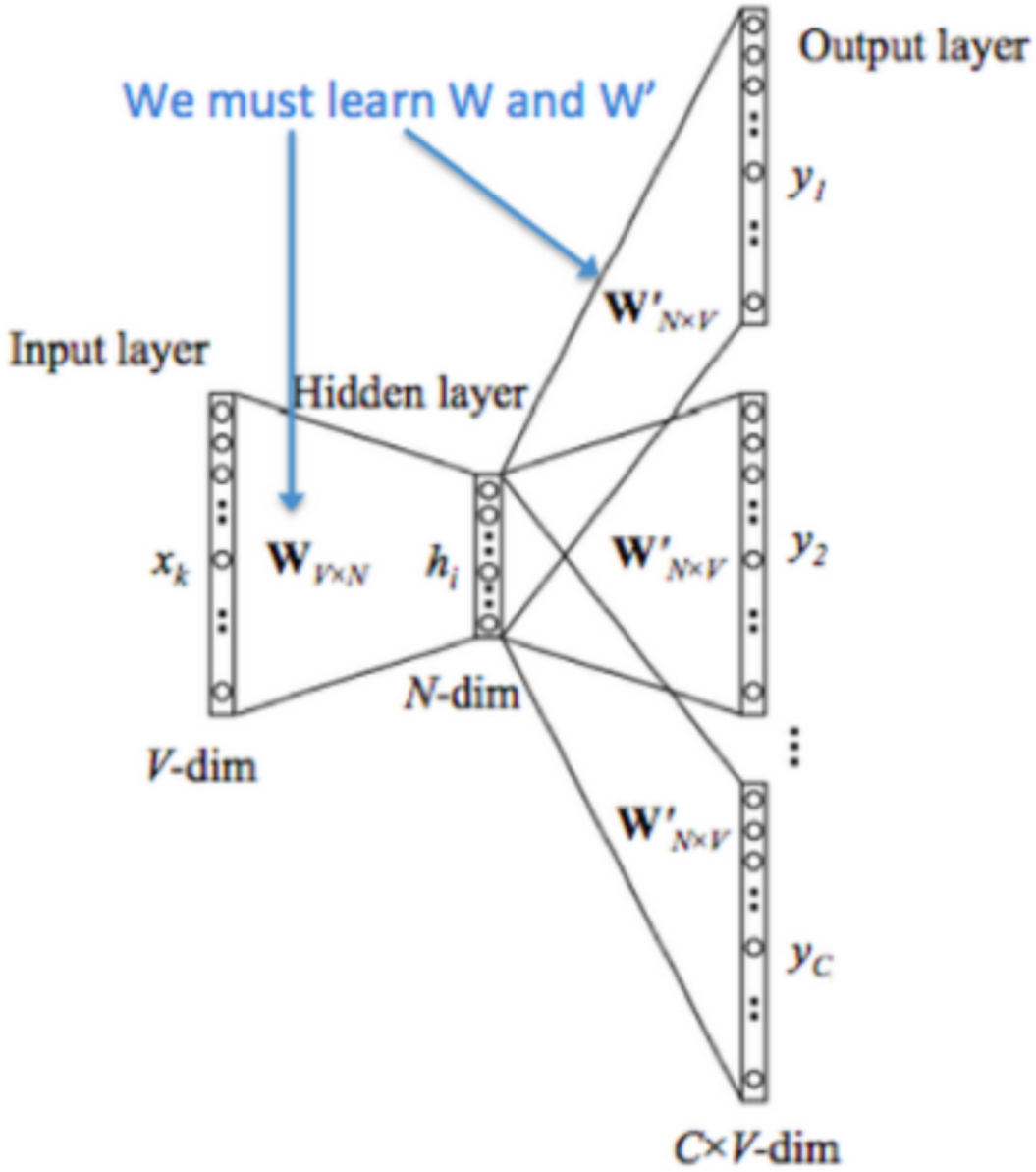


- v_{wi}和v_{wo}是输入输出向量，**W：词汇表中单词个数**
- 计算代价与W成正比，训练难度急剧增加

- 经过良好学习后的词向量可以进行线性运算

$$\overrightarrow{Madrid}-\overrightarrow{Spain}+\overrightarrow{France}=\overrightarrow{Paris}$$

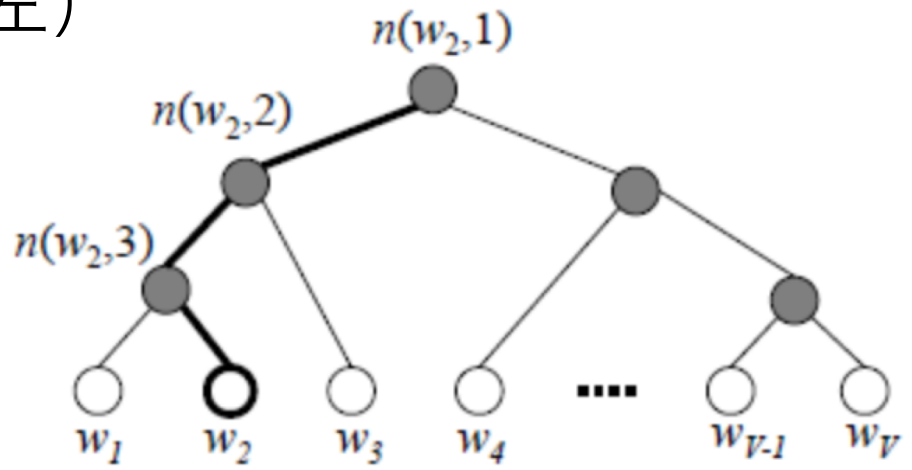
$$p(w_O|w_I) = \frac{\exp\left({v'_{w_O}}^{\top} v_{w_I}\right)}{\sum_{w=1}^{W} \exp\left({v'_w}^{\top} v_{w_I}\right)}$$



优化计算量——hierarchical softmax

- 使用binary tree表示输出层，W个词分别作为叶子节点，每个节点表示其子节点的相对概率
- 用 n(w,j) 来表示从根节点到 w 词这条路径上的第 j 个节点
- 用sigmoid函数判别正类（右）和负类（左）

$$P(+) = \sigma(x_w^T\theta) = \frac{1}{1+e^{-x_w^T\theta}}$$



贪心科技 | 让每个人享受个性化教育服务

优化计算量——hierarchical softmax

- 由于是二叉树，在输出层不需要计算 W 个节点，而只需要计算 log(W) 即可
- 分层softmax使用的树结构对性能有很大影响
- 霍夫曼树：高频的词靠近树根，这样高频词需要更少的时间会被找到——简单且高效的加速训练技术

Hierarchical softmax 缺点

- 提高模型训练的效率
- 如果我们的训练样本里的中心词w是一个很生僻的词，则耗时很久

对skip-gram的扩展
对高频词进行二次采样（sub-sampling）

- 大语料库中最频繁的单词很容易出现数亿次（**in**、**the**和**a**），这样的词通常比罕见的词提供的信息价值更少
- 为了解决低频词和高频词之间的不平衡，每个词都有一定的概率被丢弃

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}}$$

- 通过二次采样平衡数据，可以更好的学习到低频词

对skip-gram的扩展
负采样negative sampling

- noise contrastive estimation，NCE 噪声对比估计，假设一个好的模型可以通过逻辑回归来区分正常数据和噪声
- Negative sampling，NEG负采样中不是多分类而是二分类问题，给定两个词，其中一个词做Content判断另一个词是不是target，如果是则输出1，如果不是则输出0。
- 一组正取样+n组负取样（小数据集n选5-20，大数据集n取2-5）
- Logistic模型： $P(y=1|c,t)=\sigma(\theta_t^T e_c)$
- 词向量和字典中的每一个词都会进行一次二分类，来判断这个词是否是Content的Target，每次迭代只涉及k+1个单元

对skip-gram的扩展
负采样negative sampling

- Negative sampling，NEG

替换log预测概率

$$\log \sigma(v'_{w_O}{}^{\top} v_{w_I}) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} \left[\log \sigma(-v'_{w_i}{}^{\top} v_{w_I}) \right]$$

- Pn(w)是词 n 周围的噪声词分布，用逻辑回归从 k 个负例中区别出目标词汇 wO
- NCE 和 NEG 的区别在于 NCE 在计算时需要样本和噪音分布的数值概率，而 NEG 只需要样本
- 阅读 https://blog.zakjost.com/post/nce-intro/
- 代码 https://github.com/tmikolov/word2vec/blob/master/word2vec.c

如何选取负样本？

- 根据文本中单词的词频取样：常出现的词在负取样中出现次数多，模型经常做无谓的训练
- 均匀分布的随机取样：无代表性
- 折中 加权

$$P(\omega_i) = \frac{f(\omega_i)^{3/4}}{\sum_{j=1}^{n} f(\omega_j)^{3/4}}$$

- 通俗解释？噪声也要有干扰意义

实验结果

Method	Time [min]	Syntactic [%]	Semantic [%]	Total accuracy [%]
NEG-5	38	63	54	59
NEG-15	97	63	58	**61**
HS-Huffman	41	53	40	47
NCE-5	38	60	45	53
The following results use 10^{-5} subsampling				
NEG-5	14	61	58	60
NEG-15	36	61	61	**61**
HS-Huffman	21	52	59	55

692K 的词汇量，类比推理任务
句法类比(如"quick: "quickly" :: "slow":"slowly")
语义类比(如国家到首都的关系)

在实际中， hierarchical softmax 对低频词往往表现得更好，负采样对高频词和较低维度向量表现得更好。

对skip-gram的扩展
从基于word到基于phrase

修改训练前的词典部分

- 许多短语idiomatic phrase的含义并不是完全由单个单词的含义组成的 canada air；new york times
- 数据驱动的方法识别短语，组成一个 Token 作为一个词处理
  - Ngram 词表增大
  - 基于unigram bigram的方法，惩罚项用来避免太多无关的词被组合到一起

$$score(w_i, w_j) = \frac{count(w_i w_j) - \delta}{count(w_i) \times count(w_j)}$$

- 评估质量：数据集开发

$$\vec{MontrealCanadiens} - \vec{Montreal} + \vec{Toronto} = \vec{TorontoMapleLeafs}$$

  - 最好的短语表示是通过一个层次 softmax 和 Subsampling 结合的模型来学习的

词向量的可加性/合成性 A+B=?

国王相对于男人相当于女王相对于女人

好的词嵌入：
词相似度
词在整个空间的相对位置

$$\overrightarrow{\text{俄罗斯}} + \overrightarrow{\text{河}} = \overrightarrow{\text{伏尔加河}}$$

$$\overrightarrow{\text{德国}} + \overrightarrow{\text{首都}} = \overrightarrow{\text{柏林}}$$

词向量可以被看作是其上下文分布的表征；a与b相加得到向量可以看作是一个feature，只有与a和b在相同上下文经常出现的词才会接近该feature

影响因素

- 架构的选择
- 词向量的维度
- 采样率
- 训练窗口的大小

一些碎碎念

- 模型架构是食材，改进是佐料；对word2vec而言， 佐料更重要
- 怎么才能做一个好的word2vec模型？
  - 语料大，质量好

- THANKS

贪心科技 GREEDY TECHNOLOGY